

In The Claims

Please cancel claim 1 without prejudice or disclaimer of the subject matter recited therein.

Please add the following new claims:

- C29
- 1 38. A networking protocol for a network comprising:
2 a protocol packet, wherein:
3 said protocol packet is sent from a neighbor node to a node,
4 said neighbor node is a neighbor of said node,
5 said protocol packet is configured to allow said node to determine topology
6 information, and
7 said topology information comprises information regarding a topology of at
8 least a portion of said network.
 - 1 39. The networking protocol of claim 38, wherein said protocol packet comprises:
2 header data; and
3 command-specific data.
 - 1 40. The networking protocol of claim 39, wherein said header data comprises:
2 a flush indicator field.
 - 1 41. The networking protocol of claim 39, wherein said header data further
2 comprises:
3 a terminate path indicator field.
 - 1 42. The networking protocol of claim 39, wherein said header data further
2 comprises:
3 a commit path indicator field.
 - 1 43. The networking protocol of claim 39, wherein said header data comprises:
2 a request/response indicator field.

1 44. The networking protocol of claim 43, wherein said header data further
2 comprises:
3 a negative response indicator field.

C29
1 45. The networking protocol of claim 44, wherein said header data further
2 comprises:
3 a terminate path indicator field.

1 46. The networking protocol of claim 44, wherein said header data further
2 comprises:
3 a commit path indicator field.

1 47. The networking protocol of claim 44, wherein said header data comprises:
2 a flush indicator field.

1 48. The networking protocol of claim 39, wherein said protocol packet is a
2 initialization packet.

1 49. The networking protocol of claim 48, wherein said command-specific data
2 comprises:
3 information regarding a link between said node and said neighbor node.

1 50. The networking protocol of claim 48, wherein said command-specific data
2 comprises:
3 a link cost field.

1 51. The networking protocol of claim 50, wherein said command-specific data
2 further comprises:
3 a quality of service 3 capacity field; and
4 a quality of service n capacity field.

1 52. The networking protocol of claim 50, wherein said command-specific data
2 further comprises:

3 a hello interval field; and
4 a hello dead interval field.

1 53. The networking protocol of claim 39, wherein said protocol packet is a hello
2 packet.

1 54. The networking protocol of claim 53, wherein said command-specific data
2 comprises:

3 a link state advertisement count field.

1 55. The networking protocol of claim 53, wherein said command-specific data
2 further comprises:

3 an advertising node field;
4 an instance identifier field;
5 a hop count field; and
6 a neighbor count field.

1 56. The networking protocol of claim 53, wherein said command-specific data
2 further comprises:

3 a neighbor field; and
4 a link cost field.

1 57. The networking protocol of claim 53, wherein said command-specific data
2 further comprises:

3 a quality of service 3 capacity field; and
4 a quality of service n capacity field.

1 58. The networking protocol of claim 39, wherein said protocol packet is a restore
2 path packet.

1 59. The networking protocol of claim 58, wherein said command-specific data
2 comprises:

3 a virtual path identifier field.

1 60. The networking protocol of claim 59, wherein said command-specific data
2 comprises:

3 a path length field.

1 61. The networking protocol of claim 59, wherein said command-specific data
2 comprises:

3 a path index field; and
4 a path array.

1 62. The networking protocol of claim 39, wherein said protocol packet is a create
2 path packet.

1 63. The networking protocol of claim 62, wherein said command-specific data
2 comprises:

3 a virtual path identifier field;
4 a path length field;
5 a path index field; and
6 a path array.

1 64. The networking protocol of claim 39, wherein said protocol packet is a delete
2 path packet.

1 65. The networking protocol of claim 64, wherein said command-specific data
2 comprises:

3 a virtual path identifier field;
4 a path length field;
5 a path index field; and

6 a path array.

1 66. The networking protocol of claim 39, wherein said protocol packet is a test
2 path packet.

1 67. The networking protocol of claim 66, wherein said command-specific data
2 comprises:

3 a virtual path identifier field;
4 a path length field;
5 a path index field; and
6 a path array.

1 68. The networking protocol of claim 39, wherein said protocol packet is a get link
2 state advertisement packet.

1 69. The networking protocol of claim 39, wherein said protocol packet is a link
2 down packet.

1 70. The networking protocol of claim 39, wherein said protocol packet is a
2 configure packet.

1 71. A computer readable media encoded with a data structure, said data structure
2 comprising:

3 a restore path request entry, wherein said restore path request entry is maintained in
4 response to receipt of a restore path packet.

1 72. The data structure of claim 71, wherein said restore path request entry
2 comprises:

3 an origin node field containing a node id of a node that originated said restore path
4 packet; and
5 a target node field containing a node id of a target node of said restore path packet.

1 73. The data structure of claim 71, wherein said restore path request entry
2 comprises:

3 a received-from field containing a neighbor node from which said restore path packet
4 was received.

1 C29
2 74. The data structure of claim 71, wherein said restore path request entry
3 comprises:

4 a first sequence number field containing a sequence number of a first received copy of
5 said restore path packet; and
6 a last sequence number field containing a sequence number of a last received copy of
 said restore path packet.

1 75. The data structure of claim 71, wherein said restore path request entry
2 comprises:

3 a bandwidth field containing data representing a requested bandwidth; and
4 a quality of service field containing data representing a quality of service.

1 76. The data structure of claim 71, wherein said restore path request entry
2 comprises:

3 a timer field containing a timeout for said restore path packet.

1 77. The data structure of claim 71, wherein said restore path request entry
2 comprises:

3 a terminate field containing data to representing that a terminate indicator has been
4 received from a neighbor node.

1 78. The data structure of claim 71, wherein said restore path request entry
2 comprises:

3 a pending replies field containing a count of neighbor nodes that have not
4 acknowledged said restore path packet; and

5 a sent-to field containing data representing a list of neighbor nodes that have received
6 a copy of said restore path packet.

1 C30
2 79. A method of detecting a failure at a node in a network comprising:
3 periodically determining if a hello packet has been received from a neighbor node; and
4 if said hello packet has not been received from said neighbor node,
5 determining if said hello packet has been received during a period of time, and
6 if said hello packet has not been received during said period of time,
7 performing failure processing.

1 80. The method of claim 79, further comprising:
2 initializing an inactivity counter at said node;
3 if said hello packet has not been received,
4 updating said inactivity counter,
5 comparing said inactivity counter and a hello dead interval, and
6 performing failure processing, if said comparing indicates a path between said
7 node and said neighbor node has failed.

1 81. The method of claim 79, wherein said failure processing comprises:
2 changing a state of said neighbor node from an active state to a down state.

1 82. The method of claim 79, wherein said failure processing comprises:
2 changing a value of a hop count field of a link state advertisement to indicate that said
3 link state advertisement cannot be broadcast to other nodes in said network.

1 83. The method of claim 79, wherein said failure processing comprises:
2 initiating a link state advertisement removal process, wherein said link state
3 advertisement removal process removes a corresponding link state
4 advertisement from a topology database maintained at said node.

1 84. The method of claim 83, wherein said failure processing further comprises:
2 sending said link state advertisement to another neighbor node, wherein said another
3 neighbor node is a neighbor of said node.

1 C29
1 85. The method of claim 79, wherein said failure processing comprises:
2 sending a link down packet.

1 86. The method of claim 79, wherein said failure processing comprises:
2 sending a get link state advertisement packet.

1 87. A computer system comprising:
2 a processor;
3 computer readable medium coupled to said processor; and
4 computer code, encoded in said computer readable medium, configured to cause said
5 processor to:
6 periodically determine if a hello packet has been received from a neighbor
7 node; and
8 if said hello packet has not been received from said neighbor node,
9 determine if said hello packet has been received during a period of
10 time, and
11 if said hello packet has not been received during said period of time,
12 perform failure processing.

1 88. The computer system of claim 87, wherein said computer code is further
2 configured to cause said processor to:
3 initialize an inactivity counter at said node;
4 if said hello packet has not been received,
5 update said inactivity counter,
6 compare said inactivity counter and a hello dead interval, and
7 perform failure processing, if said comparing indicates a path between said
8 node and said neighbor node has failed.

1 89. The computer system of claim 87, wherein said computer code configured to
2 cause said processor to perform failure processing is further configured to cause said
3 processor to:

4 change a state of said neighbor node from an active state to a down state.

1 C29
2 90. The computer system of claim 87, wherein said computer code configured to
3 cause said processor to perform failure processing is further configured to cause said
4 processor to:

5 change a value of a hop count field of a link state advertisement to indicate that said
6 link state advertisement cannot be broadcast to other nodes in said network.

1 91. The computer system of claim 87, wherein said computer code configured to
2 cause said processor to perform failure processing is further configured to cause said
3 processor to:

4 initiate a link state advertisement removal process, wherein said link state
5 advertisement removal process removes a corresponding link state
6 advertisement from a topology database maintained at said node.

1 92. The computer system of claim 91, wherein said computer code configured to
2 cause said processor to perform failure processing is further configured to cause said
3 processor to:

4 send said link state advertisement to another neighbor node, wherein said another
5 neighbor node is a neighbor of said node.

1 93. The computer system of claim 87, wherein said computer code configured to
2 cause said processor to perform failure processing is further configured to cause said
3 processor to:

4 send a link down packet.

1 94. The computer system of claim 87, wherein said computer code configured to
2 cause said processor to perform failure processing is further configured to cause said
3 processor to:

4 send a get link state advertisement packet.

1 95. A computer program product encoded in computer readable media, said
2 computer program product comprising:

3 a first set of instructions, executable on a computer system, configured to periodically
4 determine if a hello packet has been received from a neighbor node;
5 a second set of instructions, executable on said computer system, configured to, if said
6 hello packet has not been received from said neighbor node, determine if said
7 hello packet has been received during a period of time; and
8 a third set of instructions, executable on said computer system, configured to, if said
9 hello packet has not been received from said neighbor node and said hello
10 packet has not been received during said period of time, perform failure
11 processing.

1 96. The computer program product of claim 95, further comprising:

2 a fourth set of instructions, executable on said computer system, configured to
3 initialize an inactivity counter at said node; and
4 a fifth set of instructions, executable on said computer system, configured to if said
5 hello packet has not been received,
6 update said inactivity counter,
7 compare said inactivity counter and a hello dead interval, and
8 perform failure processing, if said comparing indicates a path between said
9 node and said neighbor node has failed.

1 97. The computer program product of claim 95, wherein said third set of
2 instructions further comprises:

3 a first sub-set of instructions, executable on said computer system, configured to
4 change a state of said neighbor node from an active state to a down state.

1 98. The computer program product of claim 95, wherein said third set of
2 instructions further comprises:

3 a first sub-set of instructions, executable on said computer system, configured to
4 change a value of a hop count field of a link state advertisement to indicate that
5 said link state advertisement cannot be broadcast to other nodes in said
6 network.

1 99. The computer program product of claim 95, wherein said third set of
2 instructions further comprises:

3 a first sub-set of instructions, executable on said computer system, configured to
4 initiate a link state advertisement removal process, wherein said link state
5 advertisement removal process removes a corresponding link state
6 advertisement from a topology database maintained at said node.

1 100. The computer program product of claim 99, wherein said third set of
2 instructions further comprises:

3 a second sub-set of instructions, executable on said computer system, configured to
4 send said link state advertisement to another neighbor node, wherein said
5 another neighbor node is a neighbor of said node.

1 101. The computer program product of claim 95, wherein said third set of
2 instructions further comprises:

3 a first sub-set of instructions, executable on said computer system, configured to send
4 a link down packet.

1 102. The computer program product of claim 95, wherein said third set of
2 instructions further comprises:

3 a first sub-set of instructions, executable on said computer system, configured to send
4 a get link state advertisement packet.

1 103. An apparatus for detecting a failure at a node in a network comprising:
2 means for periodically determining if a hello packet has been received from a neighbor
3 node;

4 means for determining if said hello packet has been received during a period of time;
5 and

6 means for performing failure processing, if said hello packet has not been received
7 from said neighbor node and if said hello packet has not been received during
8 said period of time.

1 104. The apparatus of claim 103, further comprising:

2 means for initializing an inactivity counter at said node;

3 means for updating said inactivity counter, if said hello packet has not been received,
4 means for comparing said inactivity counter and a hello dead interval, if said hello
5 packet has not been received, and

6 means for performing failure processing, if said hello packet has not been received and
7 if said comparing indicates a path between said node and said neighbor node
8 has failed.

1 105. The apparatus of claim 103, wherein said means for performing failure
2 processing comprises:

3 means for changing a state of said neighbor node from an active state to a down state.

C29
1 106. The apparatus of claim 103, wherein said means for performing failure
2 processing comprises:

3 means for changing a value of a hop count field of a link state advertisement to
4 indicate that said link state advertisement cannot be broadcast to other nodes in
5 said network.

1 107. The apparatus of claim 103, wherein said means for performing failure
2 processing comprises:

3 means for initiating a link state advertisement removal process, wherein said link state
4 advertisement removal process removes a corresponding link state
5 advertisement from a topology database maintained at said node.

1 108. The apparatus of claim 107, wherein said means for performing failure
2 processing further comprises:

3 means for sending said link state advertisement to another neighbor node, wherein said
4 another neighbor node is a neighbor of said node.

1 109. The apparatus of claim 103, wherein said means for performing failure
2 processing comprises:

3 means for sending a link down packet.

1 110. The apparatus of claim 103, wherein said means for performing failure
2 processing comprises:

3 means for sending a get link state advertisement packet.

1 111. A method of processing a get link state advertisement packet comprising:
2 receiving said get link state advertisement packet at a downstream node, wherein
3 said get link state advertisement packet is sent by a sending node,
4 said get link state advertisement packet comprises at least one node identifier,
5 said at least one node identifier identifies a node in a network for which said
6 sending node seeks a link state advertisement, and

7 said downstream node and said sending node are nodes in said network; and
8 sending at least one link state advertisement to said node.

1 C29
2 112. The method of claim 111, further comprising:
3 sending an acknowledgement to said downstream node.

1 113. The method of claim 111, further comprising:
2 building a first list from a link state database maintained at said downstream node,
3 wherein
4 said first list comprises any link state advertisements received from a node
5 other than said sending node, and
6 said at least one link state advertisement is among said any link state
7 advertisements received from said sending node.

1 114. The method of claim 113, further comprising:
2 building a second list from said link state database, wherein
3 said second list comprises any link state advertisements received from said
4 sending node.

1 115. The method of claim 114, further comprising:
2 sending a get link state advertisement packet to each node corresponding to one of
3 said link state advertisements in said second list.

1 116. The method of claim 114, further comprising:
2 indicating link state advertisements in said second list are to be deleted.

1 117. The method of claim 116, further comprising:
2 deleting each one of said link state advertisements in said second list, if an updated
3 link state advertisement is not received within a period of time.

1 118. The method of claim 111, further comprising:
2 identifying said at least one link state advertisement in a link state database maintained
3 at said downstream node using said at least one node identifier.

C29
1 119. The method of claim 118, further comprising:
2 building a first list from said link state database, wherein
3 said first list comprises any link state advertisements received from a node
4 other than said sending node, and
5 said at least one link state advertisement is among said any link state
6 advertisements received from said sending node.

1 120. The method of claim 119, further comprising:
2 building a second list from said link state database, wherein
3 said second list comprises any link state advertisements received from said
4 sending node.

1 121. The method of claim 120, further comprising:
2 sending a get link state advertisement packet to each node corresponding to one of
3 said link state advertisements in said second list.

1 122. The method of claim 120, further comprising:
2 indicating link state advertisements in said second list are to be deleted.

1 123. The method of claim 122, further comprising:
2 deleting each one of said link state advertisements in said second list, if an updated
3 link state advertisement is not received within a period of time.

1 124. A computer system comprising:
2 a processor;
3 computer readable medium coupled to said processor; and

4 computer code, encoded in said computer readable medium, configured to cause said
5 processor to:
6 receive said get link state advertisement packet at a downstream node, wherein
7 said get link state advertisement packet is sent by a sending node,
8 said get link state advertisement packet comprises at least one node
9 identifier,
10 said at least one node identifier identifies a node in a network for which
11 said sending node seeks a link state advertisement, and
12 said downstream node and said sending node are nodes in said network;
13 and
14 send at least one link state advertisement to said node.

1 125. The computer system of claim 124, wherein said computer code is further
2 configured to cause said processor to:
3 send an acknowledgement to said downstream node.

1 126. The computer system of claim 124, wherein said computer code is further
2 configured to cause said processor to:
3 build a first list from a link state database maintained at said downstream node,
4 wherein
5 said first list comprises any link state advertisements received from a node
6 other than said sending node, and
7 said at least one link state advertisement is among said any link state
8 advertisements received from said sending node.

1 127. The computer system of claim 126, wherein said computer code is further
2 configured to cause said processor to:
3 build a second list from said link state database, wherein
4 said second list comprises any link state advertisements received from said
5 sending node.

1 128. The computer system of claim 127, wherein said computer code is further
2 configured to cause said processor to:

3 send a get link state advertisement packet to each node corresponding to one of said
4 link state advertisements in said second list.

1 C29
2 129. The computer system of claim 127, wherein said computer code is further
3 configured to cause said processor to:

4 indicate link state advertisements in said second list are to be deleted.

1 130. The computer system of claim 129, wherein said computer code is further
2 configured to cause said processor to:

3 delete each one of said link state advertisements in said second list, if an updated link
4 state advertisement is not received within a period of time.

1 131. The computer system of claim 124, wherein said computer code is further
2 configured to cause said processor to:

3 identify said at least one link state advertisement in a link state database maintained at
4 said downstream node using said at least one node identifier.

1 132. The computer system of claim 131, wherein said computer code is further
2 configured to cause said processor to:

3 build a first list from said link state database, wherein
4 said first list comprises any link state advertisements received from a node
5 other than said sending node, and
6 said at least one link state advertisement is among said any link state
7 advertisements received from said sending node.

1 133. The computer system of claim 132, wherein said computer code is further
2 configured to cause said processor to:

3 build a second list from said link state database, wherein

4 said second list comprises any link state advertisements received from said
5 sending node.

1 C29
2 134. The computer system of claim 133, wherein said computer code is further
configured to cause said processor to:

3 send a get link state advertisement packet to each node corresponding to one of said
4 link state advertisements in said second list.

1 135. The computer system of claim 133, wherein said computer code is further
2 configured to cause said processor to:

3 indicate link state advertisements in said second list are to be deleted.

1 136. The computer system of claim 135, wherein said computer code is further
2 configured to cause said processor to:

3 deleting each one of said link state advertisements in said second list, if an updated
4 link state advertisement is not received within a period of time.

1 137. A computer program product encoded in computer readable media, said
2 computer program product comprising:

3 a first set of instructions, executable on a computer system, configured to receive said
4 get link state advertisement packet at a downstream node, wherein
5 said get link state advertisement packet is sent by a sending node,
6 said get link state advertisement packet comprises at least one node identifier,
7 said at least one node identifier identifies a node in a network for which said
8 sending node seeks a link state advertisement, and
9 said downstream node and said sending node are nodes in said network; and
10 a second set of instructions, executable on said computer system, configured to send at
11 least one link state advertisement to said node.

1 138. The computer program product of claim 137, further comprising:

2 a third set of instructions, executable on said computer system, configured to send an
3 acknowledgement to said downstream node.

1 139. The computer program product of claim 137, further comprising:
2 a third set of instructions, executable on said computer system, configured to build a
3 first list from a link state database maintained at said downstream node,
4 wherein

5 said first list comprises any link state advertisements received from a node
6 other than said sending node, and
7 said at least one link state advertisement is among said any link state
8 advertisements received from said sending node.

1 140. The computer program product of claim 139, further comprising:
2 a fourth set of instructions, executable on said computer system, configured to build a
3 second list from said link state database, wherein
4 said second list comprises any link state advertisements received from said
5 sending node.

1 141. The computer program product of claim 140, further comprising:
2 a fifth set of instructions, executable on said computer system, configured to send a
3 get link state advertisement packet to each node corresponding to one of said
4 link state advertisements in said second list.

1 142. The computer program product of claim 140, further comprising:
2 a fifth set of instructions, executable on said computer system, configured to indicate
3 link state advertisements in said second list are to be deleted.

1 143. The computer program product of claim 142, further comprising:
2 a sixth set of instructions, executable on said computer system, configured to delete
3 each one of said link state advertisements in said second list, if an updated link
4 state advertisement is not received within a period of time.

C 29

1 144. The computer program product of claim 137, further comprising:
2 a third set of instructions, executable on said computer system, configured to identify
3 said at least one link state advertisement in a link state database maintained at
4 said downstream node using said at least one node identifier.

1 145. The computer program product of claim 144, further comprising:
2 a fourth set of instructions, executable on said computer system, configured to build a
3 first list from said link state database, wherein
4 said first list comprises any link state advertisements received from a node
5 other than said sending node, and
6 said at least one link state advertisement is among said any link state
7 advertisements received from said sending node.

1 146. The computer program product of claim 145, further comprising:
2 a fifth set of instructions, executable on said computer system, configured to build a
3 second list from said link state database, wherein
4 said second list comprises any link state advertisements received from said
5 sending node.

1 147. The computer program product of claim 146, further comprising:
2 a sixth set of instructions, executable on said computer system, configured to send a
3 get link state advertisement packet to each node corresponding to one of said
4 link state advertisements in said second list.

1 148. The computer program product of claim 146, further comprising:
2 a sixth set of instructions, executable on said computer system, configured to indicate
3 link state advertisements in said second list are to be deleted.

C29

1 149. The computer program product of claim 148, further comprising:
2 a seventh set of instructions, executable on said computer system, configured to
3 deleting each one of said link state advertisements in said second list, if an
4 updated link state advertisement is not received within a period of time.

1 150. An apparatus for processing a get link state advertisement packet comprising:
2 means for receiving said get link state advertisement packet at a downstream node,
3 wherein
4 said get link state advertisement packet is sent by a sending node,
5 said get link state advertisement packet comprises at least one node identifier,
6 said at least one node identifier identifies a node in a network for which said
7 sending node seeks a link state advertisement, and
8 said downstream node and said sending node are nodes in said network; and
9 means for sending at least one link state advertisement to said node.

1 151. The apparatus of claim 150, further comprising:
2 means for sending an acknowledgement to said downstream node.

1 152. The apparatus of claim 150, further comprising:
2 means for building a first list from a link state database maintained at said downstream
3 node, wherein
4 said first list comprises any link state advertisements received from a node
5 other than said sending node, and
6 said at least one link state advertisement is among said any link state
7 advertisements received from said sending node.

1 153. The apparatus of claim 152, further comprising:
2 means for building a second list from said link state database, wherein
3 said second list comprises any link state advertisements received from said
4 sending node.

C29

- 1 154. The apparatus of claim 153, further comprising:
2 means for sending a get link state advertisement packet to each node corresponding to
3 one of said link state advertisements in said second list.

- 1 155. The apparatus of claim 153, further comprising:
2 means for indicating link state advertisements in said second list are to be deleted.

- 1 156. The apparatus of claim 155, further comprising:
2 means for deleting each one of said link state advertisements in said second list, if an
3 updated link state advertisement is not received within a period of time.

- 1 157. The apparatus of claim 150, further comprising:
2 means for identifying said at least one link state advertisement in a link state database
3 maintained at said downstream node using said at least one node identifier.

- 1 158. The apparatus of claim 157, further comprising:
2 means for building a first list from said link state database, wherein
3 said first list comprises any link state advertisements received from a node
4 other than said sending node, and
5 said at least one link state advertisement is among said any link state
6 advertisements received from said sending node.

- 1 159. The apparatus of claim 158, further comprising:
2 means for building a second list from said link state database, wherein
3 said second list comprises any link state advertisements received from said
4 sending node.

- 1 160. The apparatus of claim 159, further comprising:
2 means for sending a get link state advertisement packet to each node corresponding to
3 one of said link state advertisements in said second list.

- C29
- 1 161. The apparatus of claim 159, further comprising:
2 means for indicating link state advertisements in said second list are to be deleted.
 - 1 162. The apparatus of claim 161, further comprising:
2 means for deleting each one of said link state advertisements in said second list, if an
3 updated link state advertisement is not received within a period of time.
 - 1 163. A method of processing a get link state advertisement packet comprising:
2 receiving a hello packet at a downstream node, wherein said hello packet comprises a
3 link state advertisement; and
4 processing said link state advertisement.
 - 1 164. The method of claim 163, further comprising:
2 sending an acknowledgement to said downstream node, wherein said
3 acknowledgement acknowledges all link state advertisements in said hello
4 packet.
 - 1 165. The method of claim 163, wherein said processing comprises:
2 determining if said link state advertisement corresponds to an entry in a link state
3 database maintained at said downstream node.
 - 1 166. The method of claim 165, wherein said processing further comprises:
2 if said link state advertisement does not correspond to an entry in a link state database
3 maintained at said downstream node,
4 adding said link state advertisement to said link state database.
 - 1 167. The method of claim 166, wherein said processing further comprises:
2 if said link state advertisement corresponds to an entry in a link state database
3 maintained at said downstream node,

4 determining if a node originating said link state advertisement is a node
5 originating a link state advertisement corresponding to said entry in
6 said link state database.

1 C29
2 168. The method of claim 167, wherein said processing further comprises:
3 if said node originating said link state advertisement is not said node originating said
4 link state advertisement corresponding to said entry in said link state database,
5 adding said link state advertisement to said link state database.

1 169. The method of claim 167, wherein said processing further comprises:
2 if said node originating said link state advertisement is said node originating said link
3 state advertisement corresponding to said entry in said link state database,
4 determining if said link state advertisement is more recent than said link state
5 advertisement corresponding to said entry in said link state database.

1 170. The method of claim 169, wherein said processing further comprises:
2 if said link state advertisement is not more recent than said link state advertisement
3 corresponding to said entry in said link state database,
4 discarding said link state advertisement.

1 171. The method of claim 169, wherein said processing further comprises:
2 if said link state advertisement is more recent than said link state advertisement
3 corresponding to said entry in said link state database,
4 adding said link state advertisement to said link state database.

1 172. The method of claim 169, wherein said determining if said link state
2 advertisement is more recent than said link state advertisement corresponding to said entry in
3 said link state database comprises:
4 determining if a link state identifier of said link state advertisement is the same as a
5 link state identifier of said link state advertisement corresponding to said entry
6 in said link state database.

C29
1 173. The method of claim 172, wherein said determining if said link state
2 advertisement is more recent than said link state advertisement corresponding to said entry in
3 said link state database further comprises:

4 if said link state identifier of said link state advertisement is not the same as said link
5 state identifier of said link state advertisement corresponding to said entry in
6 said link state database,
7 indicating a one of said link state advertisement and said link state
8 advertisement corresponding to said entry in said link state database
9 having a higher link state identifier is more recent.

1 174. The method of claim 172, wherein said determining if said link state
2 advertisement is more recent than said link state advertisement corresponding to said entry in
3 said link state database further comprises:

4 if said link state identifier of said link state advertisement is the same as said link state
5 identifier of said link state advertisement corresponding to said entry in said
6 link state database,
7 determining if a hop count of said link state advertisement is the same as a hop
8 count of said link state advertisement corresponding to said entry in
9 said link state database.

1 175. The method of claim 174, wherein said determining if said link state
2 advertisement is more recent than said link state advertisement corresponding to said entry in
3 said link state database further comprises:

4 if said hop count of said link state advertisement is the same as said hop count of said
5 link state advertisement corresponding to said entry in said link state database,
6 indicating that said link state advertisement and said link state advertisement
7 corresponding to said entry in said link state database are the same.

C29
1 176. The method of claim 174, wherein said determining if said link state
2 advertisement is more recent than said link state advertisement corresponding to said entry in
3 said link state database further comprises:

4 if said hop count of said link state advertisement is not the same as said hop count of
5 said link state advertisement corresponding to said entry in said link state
6 database,
7 indicating that the one of said link state advertisement and said link state
8 advertisement corresponding to said entry in said link state database
9 having a lower hop count is more recent.

1 177. A computer system comprising:
2 a processor;
3 computer readable medium coupled to said processor; and
4 computer code, encoded in said computer readable medium, configured to cause said
5 processor to:
6 receive a hello packet at a downstream node, wherein said hello packet
7 comprises a link state advertisement; and
8 process said link state advertisement.

1 178. The computer system of claim 177, wherein said computer code is further
2 configured to cause said processor to:
3 send an acknowledgement to said downstream node, wherein said acknowledgement
4 acknowledges all link state advertisements in said hello packet.

1 179. The computer system of claim 177, wherein said computer code configured to
2 cause said processor to process said link state advertisement is further configured to cause
3 said processor to:
4 determine if said link state advertisement corresponds to an entry in a link state
5 database maintained at said downstream node.

C29
1 180. The computer system of claim 179, wherein said computer code configured to
2 cause said processor to process said link state advertisement is further configured to cause
3 said processor to:

4 if said link state advertisement does not correspond to an entry in a link state database
5 maintained at said downstream node,
6 add said link state advertisement to said link state database.

1 181. The computer system of claim 180, wherein said computer code configured to
2 cause said processor to process said link state advertisement is further configured to cause
3 said processor to:

4 if said link state advertisement corresponds to an entry in a link state database
5 maintained at said downstream node,
6 determine if a node originating said link state advertisement is a node
7 originating a link state advertisement corresponding to said entry in
8 said link state database.

1 182. The computer system of claim 181, wherein said computer code configured to
2 cause said processor to process said link state advertisement is further configured to cause
3 said processor to:

4 if said node originating said link state advertisement is not said node originating said
5 link state advertisement corresponding to said entry in said link state database,
6 add said link state advertisement to said link state database.

1 183. The computer system of claim 181, wherein said computer code configured to
2 cause said processor to process said link state advertisement is further configured to cause
3 said processor to:

4 if said node originating said link state advertisement is said node originating said link
5 state advertisement corresponding to said entry in said link state database,
6 determine if said link state advertisement is more recent than said link state
7 advertisement corresponding to said entry in said link state database.

1 184. The computer system of claim 183, wherein said computer code configured to
2 cause said processor to process said link state advertisement is further configured to cause
3 said processor to:

4 if said link state advertisement is not more recent than said link state advertisement
5 corresponding to said entry in said link state database,
6 discard said link state advertisement.

1 185. The computer system of claim 183, wherein said computer code configured to
2 cause said processor to process said link state advertisement is further configured to cause
3 said processor to:

4 if said link state advertisement is more recent than said link state advertisement
5 corresponding to said entry in said link state database,
6 add said link state advertisement to said link state database.

1 186. The computer system of claim 183, wherein said computer code configured to
2 cause said processor to process said link state advertisement is further configured to cause
3 said processor to:

4 determine if a link state identifier of said link state advertisement is the same as a link
5 state identifier of said link state advertisement corresponding to said entry in
6 said link state database.

1 187. The computer system of claim 186, wherein said computer code configured to
2 cause said processor to determine if said link state advertisement is more recent than said link
3 state advertisement corresponding to said entry in said link state database is further configured
4 to cause said processor to:

5 if said link state identifier of said link state advertisement is not the same as said link
6 state identifier of said link state advertisement corresponding to said entry in
7 said link state database,

C29
8 indicate a one of said link state advertisement and said link state advertisement
9 corresponding to said entry in said link state database having a higher
10 link state identifier is more recent.

1 188. The computer system of claim 186, wherein said computer code configured to
2 cause said processor to determine if said link state advertisement is more recent than said link
3 state advertisement corresponding to said entry in said link state database is further configured
4 to cause said processor to:

5 if said link state identifier of said link state advertisement is the same as said link state
6 identifier of said link state advertisement corresponding to said entry in said
7 link state database,
8 determine if a hop count of said link state advertisement is the same as a hop
9 count of said link state advertisement corresponding to said entry in
10 said link state database.

1 189. The computer system of claim 188, wherein said computer code configured to
2 cause said processor to determine if said link state advertisement is more recent than said link
3 state advertisement corresponding to said entry in said link state database is further configured
4 to cause said processor to:

5 if said hop count of said link state advertisement is the same as said hop count of said
6 link state advertisement corresponding to said entry in said link state database,
7 indicate that said link state advertisement and said link state advertisement
8 corresponding to said entry in said link state database are the same.

1 190. The computer system of claim 188, wherein said computer code configured to
2 cause said processor to determine if said link state advertisement is more recent than said link
3 state advertisement corresponding to said entry in said link state database is further configured
4 to cause said processor to:

5 if said hop count of said link state advertisement is not the same as said hop count of
6 said link state advertisement corresponding to said entry in said link state
7 database,

C29
8 indicating that the one of said link state advertisement and said link state
9 advertisement corresponding to said entry in said link state database
10 having a lower hop count is more recent.

- 1 191. A computer program product encoded in computer readable media, said
2 computer program product comprising:
3 a first set of instructions, executable on a computer system, configured to receive a
4 hello packet at a downstream node, wherein said hello packet comprises a link
5 state advertisement; and
6 a second set of instructions, executable on said computer system, configured to
7 process said link state advertisement.
- 1 192. The computer program product of claim 191, further comprising:
2 a third set of instructions, executable on said computer system, configured to send an
3 acknowledgement to said downstream node, wherein said acknowledgement
4 acknowledges all link state advertisements in said hello packet.
- 1 193. The computer program product of claim 191, wherein said second set of
2 instructions comprises:
3 a first sub-set of instructions, executable on said computer system, configured to
4 determine if said link state advertisement corresponds to an entry in a link state
5 database maintained at said downstream node.
- 1 194. The computer program product of claim 193, wherein said second set of
2 instructions further comprises:
3 a second sub-set of instructions, executable on said computer system, configured to, if
4 said link state advertisement does not correspond to an entry in a link state
5 database maintained at said downstream node,
6 add said link state advertisement to said link state database.

C29
1 195. The computer program product of claim 194, wherein said second set of
2 instructions further comprises:

3 a third sub-set of instructions, executable on said computer system, configured to, if
4 said link state advertisement corresponds to an entry in a link state database
5 maintained at said downstream node,
6 determine if a node originating said link state advertisement is a node
7 originating a link state advertisement corresponding to said entry in
8 said link state database.

1 196. The computer program product of claim 195, wherein said second set of
2 instructions further comprises:

3 a fourth sub-set of instructions, executable on said computer system, configured to, if
4 said node originating said link state advertisement is not said node originating
5 said link state advertisement corresponding to said entry in said link state
6 database,
7 add said link state advertisement to said link state database.

1 197. The computer program product of claim 195, wherein said second set of
2 instructions further comprises:

3 a fourth sub-set of instructions, executable on said computer system, configured to, if
4 said node originating said link state advertisement is said node originating said
5 link state advertisement corresponding to said entry in said link state database,
6 determine if said link state advertisement is more recent than said link state
7 advertisement corresponding to said entry in said link state database.

1 198. The computer program product of claim 197, wherein said second set of
2 instructions further comprises:

3 a fifth sub-set of instructions, executable on said computer system, configured to, if
4 said link state advertisement is not more recent than said link state
5 advertisement corresponding to said entry in said link state database,

6 discard said link state advertisement.

1 C29
199. The computer program product of claim 197, wherein said second set of
2 instructions further comprises:

3 a sixth sub-set of instructions, executable on said computer system, configured to, if
4 said link state advertisement is more recent than said link state advertisement
5 corresponding to said entry in said link state database,
6 add said link state advertisement to said link state database.

1 200. The computer program product of claim 197, wherein said second set of
2 instructions further comprises:

3 a sixth sub-set of instructions, executable on said computer system, configured to
4 determine if a link state identifier of said link state advertisement is the same
5 as a link state identifier of said link state advertisement corresponding to said
6 entry in said link state database.

1 201. The computer program product of claim 200, wherein said fourth sub-set of
2 instructions further comprises:

3 a first sub-sub-set of instructions, executable on said computer system, configured to,
4 if said link state identifier of said link state advertisement is not the same as
5 said link state identifier of said link state advertisement corresponding to said
6 entry in said link state database,
7 indicate a one of said link state advertisement and said link state advertisement
8 corresponding to said entry in said link state database having a higher
9 link state identifier is more recent.

1 202. The computer program product of claim 200, wherein said fourth sub-set of
2 instructions further comprises:

3 a first sub-sub-set of instructions, executable on said computer system, configured to,
4 if said link state identifier of said link state advertisement is the same as said

5 link state identifier of said link state advertisement corresponding to said entry
6 in said link state database,
7 determine if a hop count of said link state advertisement is the same as a hop
8 count of said link state advertisement corresponding to said entry in
9 said link state database.

1 203. The computer program product of claim 202, wherein said fourth sub-set of
2 instructions further comprises:

3 a second sub-sub-set of instructions, executable on said computer system, configured
4 to, if said hop count of said link state advertisement is the same as said hop
5 count of said link state advertisement corresponding to said entry in said link
6 state database,
7 indicate that said link state advertisement and said link state advertisement
8 corresponding to said entry in said link state database are the same.

1 204. The computer program product of claim 202, wherein said fourth sub-set of
2 instructions further comprises:

3 a second sub-sub-set of instructions, executable on said computer system, configured
4 to, if said hop count of said link state advertisement is not the same as said hop
5 count of said link state advertisement corresponding to said entry in said link
6 state database,
7 indicating that the one of said link state advertisement and said link state
8 advertisement corresponding to said entry in said link state database
9 having a lower hop count is more recent.

1 205. An apparatus for processing a get link state advertisement packet comprising:
2 means for receiving a hello packet at a downstream node, wherein said hello packet
3 comprises a link state advertisement; and
4 means for processing said link state advertisement.

1 206. The apparatus of claim 205, further comprising:
2 means for sending an acknowledgement to said downstream node, wherein said
3 acknowledgement acknowledges all link state advertisements in said hello
4 packet.

1 207. The apparatus of claim 205, wherein said means for processing comprises:
2 means for determining if said link state advertisement corresponds to an entry in a link
3 state database maintained at said downstream node.

1 208. The apparatus of claim 207, wherein said means for processing further
2 comprises:

3 means for adding said link state advertisement to a link state database, if said link state
4 advertisement does not correspond to an entry in a link state database
5 maintained at said downstream node.

1 209. The apparatus of claim 208, wherein said means for processing further
2 comprises:

3 means for determining if a node originating said link state advertisement is a node
4 originating a link state advertisement corresponding to an entry in a link state
5 database, if said link state advertisement corresponds to said entry in said link
6 state database maintained at said downstream node.

1 210. The apparatus of claim 209, wherein said means for processing further
2 comprises:

3 means for adding said link state advertisement to said link state database, if said node
4 originating said link state advertisement is not said node originating said link
5 state advertisement corresponding to said entry in said link state database.

1 211. The apparatus of claim 209, wherein said means for processing further
2 comprises:

3 means for determining if said link state advertisement is more recent than said link
4 state advertisement corresponding to said entry in said link state database, if
5 said node originating said link state advertisement is said node originating said
6 link state advertisement corresponding to said entry in said link state database.

C29

1 212. The apparatus of claim 211, wherein said means for processing further
2 comprises:

3 means for discarding said link state advertisement, if said link state advertisement is
4 not more recent than said link state advertisement corresponding to said entry
5 in said link state database.

1 213. The apparatus of claim 211, wherein said menas for processing further
2 comprises:

3 means for adding said link state advertisement to said link state database, if said link
4 state advertisement is more recent than said link state advertisement
5 corresponding to said entry in said link state database.

1 214. The apparatus of claim 211, wherein said means for determining if said link
2 state advertisement is more recent than said link state advertisement corresponding to said
3 entry in said link state database comprises:

4 means for determining if a link state identifier of said link state advertisement is the
5 same as a link state identifier of said link state advertisement corresponding to
6 said entry in said link state database.

1 215. The apparatus of claim 214, wherein said means for determining if said link
2 state advertisement is more recent than said link state advertisement corresponding to said
3 entry in said link state database further comprises:

4 means for indicating a one of said link state advertisement and said link state
5 advertisement corresponding to said entry in said link state database having a

6 higher link state identifier is more recent, if said link state identifier of said link
7 state advertisement is not the same as said link state identifier of said link state
8 advertisement corresponding to said entry in said link state database.

C29
1 216. The apparatus of claim 214, wherein said means for determining if said link
2 state advertisement is more recent than said link state advertisement corresponding to said
3 entry in said link state database further comprises:

4 means for determining if a hop count of said link state advertisement is the same as a
5 hop count of said link state advertisement corresponding to said entry in said
6 link state database, if said link state identifier of said link state advertisement is
7 the same as said link state identifier of said link state advertisement
8 corresponding to said entry in said link state database.

1 217. The apparatus of claim 216, wherein said means for determining if said link
2 state advertisement is more recent than said link state advertisement corresponding to said
3 entry in said link state database further comprises:

4 means for indicating that said link state advertisement and said link state
5 advertisement corresponding to said entry in said link state database are the
6 same, if said hop count of said link state advertisement is the same as said hop
7 count of said link state advertisement corresponding to said entry in said link
8 state database.

1 218. The apparatus of claim 216, wherein said means for determining if said link
2 state advertisement is more recent than said link state advertisement corresponding to said
3 entry in said link state database further comprises:

4 means for indicating that the one of said link state advertisement and said link state
5 advertisement corresponding to said entry in said link state database having a
6 lower hop count is more recent, if said hop count of said link state
7 advertisement is not the same as said hop count of said link state advertisement
8 corresponding to said entry in said link state database.